Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Інститут комп’ютерних наук та інформаційних технологій

Кафедра програмного забезпечення



**ЗВІТ**

**Про виконання практичної роботи № 4**

**з дисципліни «Програмне забезпечення мережевих технологій»**

**на тему: «Побудова програм потоків за допомогою стандартних бібліотек C#»**

**Лектор:**

проф. кафедри ПЗ

Мельник Р.А.

**Виконав:**

студ. групи ПІ-31

Павленчик М. М.

**Прийняв:**

проф. кафедри ПЗ

Мельник Р.А.

«\_\_\_» \_\_\_\_\_\_ 2017 р.

∑ = \_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_

Львів – 2017

**ТЕМА РОБОТИ**: Побудова програм потоків за допомогою стандартних бібліотек C#.

**Завдання роботи**

Ознайомитись із багатопоточністю в С#, створити застосунок, що демонструє роботу декількох потоків.

**Результати виконання**

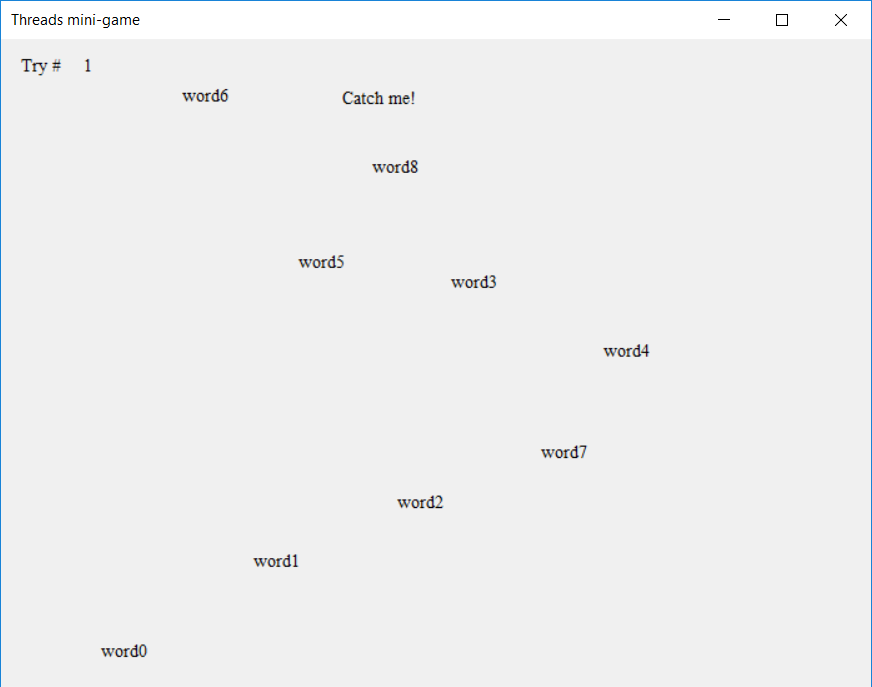


Рис. 1. Початок роботи застосунку

При запуску застосунку на формі розміщуються компоненти Label. Для кожної даної компоненти формується об’єкт класу Word, що відповідає за переміщення компоненти, яке виконується в окремому потоці. Користувач повинен натиснути на фразу «Catch me!», щоб перемогти у міні-грі. При кожному невлучному кліку лічильник буде збільшуватись на 1, і це відображатиметься у лівому верхньому куті форми.

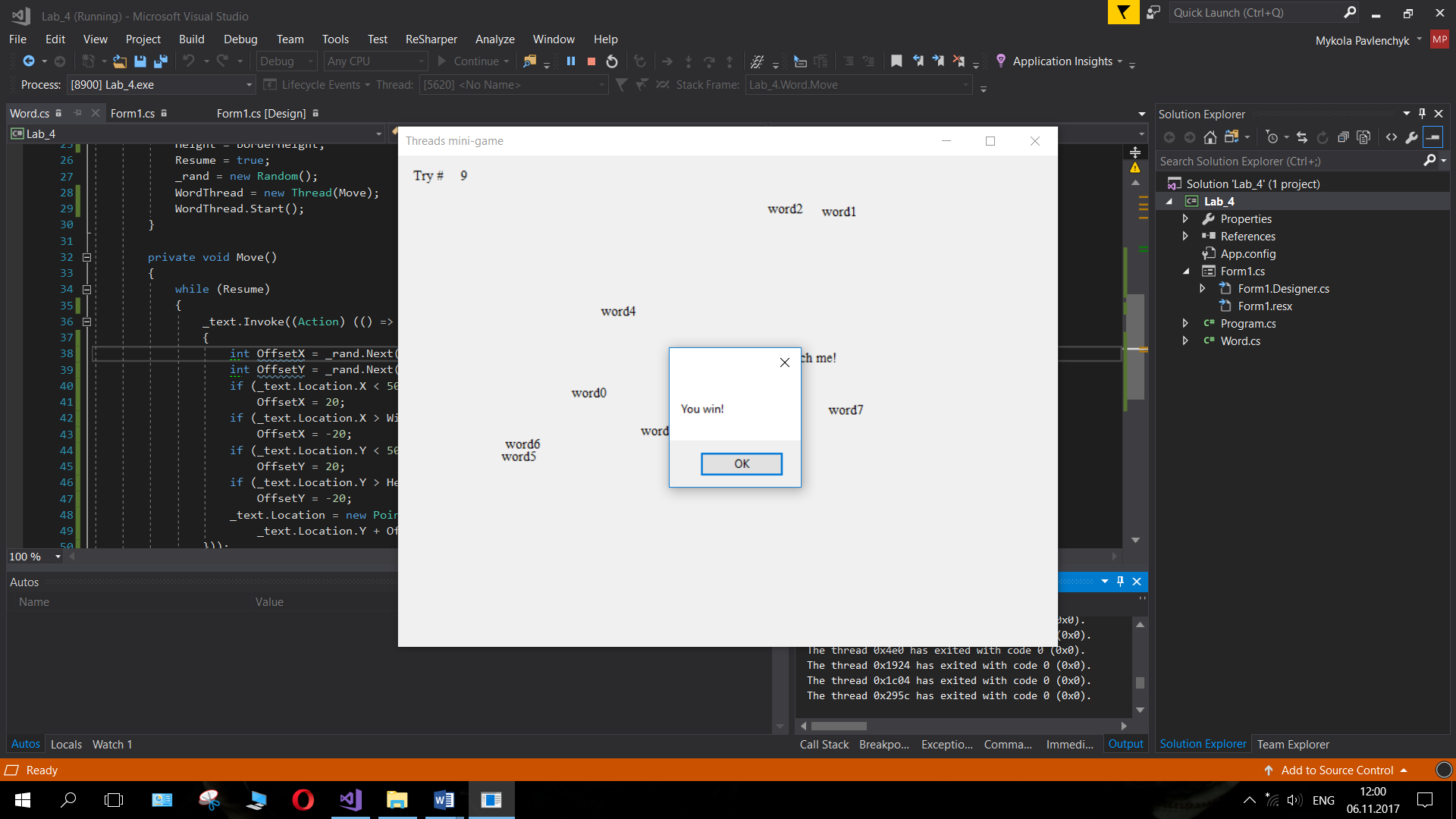


Рис. 2. Перемога у міні-грі

Вихідний код

//Код класу для роботи із окремим словом

//в окремому потоці

public class Word

{

private Label \_text;

private Random \_rand;

public int Width;

public int Height;

public bool Resume;

public Thread WordThread;

public Word(Label text, int borderWidth, int borderHeight)

{

\_text = text;

Width = borderWidth;

Height = borderHeight;

Resume = true;

\_rand = new Random();

WordThread = new Thread(Move);

WordThread.Start();

}

private void Move()

{

while (Resume)

{

\_text.Invoke((Action) (() =>

{

int OffsetX = \_rand.Next(-\_rand.Next(5, 30), \_rand.Next(5, 30));

int OffsetY = \_rand.Next(-\_rand.Next(5, 30), \_rand.Next(5, 30));

if (\_text.Location.X < 50)

OffsetX = 20;

if (\_text.Location.X > Width - 100)

OffsetX = -20;

if (\_text.Location.Y < 50)

OffsetY = 20;

if (\_text.Location.Y > Height - 50)

OffsetY = -20;

\_text.Location = new Point(\_text.Location.X + OffsetX,

\_text.Location.Y + OffsetY);

}));

Thread.Sleep(140);

}

}

}

//Основний код форми

public partial class MainForm : Form

{

private List<Label> \_words;

private List<Word> \_wordsControllers;

private Word \_winWord;

private Random \_rand;

private int \_tryCounter = 1;

public MainForm()

{

InitializeComponent();

\_rand = new Random();

\_words = new List<Label>();

\_wordsControllers = new List<Word>();

for (int i = 0; i < 10; ++i)

{

\_words.Add(new Label());

if (i < 9)

{

\_words[i].Text = "word" + i;

\_words[i].Size = new Size(50, 20);

\_words[i].Click += (sender, args) =>

{

\_tryCounter++;

TryCounterValue.Text = \_tryCounter.ToString();

};

}

else

{

\_words[i].Text = "Catch me!";

\_words[i].Size = new Size(70, 20);

\_words[i].Click += (sender, args) =>

{

\_wordsControllers.ForEach(elem => elem.Resume = false);

MessageBox.Show("You win!");

RestartButton.Visible = true;

};

}

\_words[i].Parent = this;

\_words[i].Location = new Point(\_rand.Next(50, Width - 100), \_rand.Next(50, Height - 50));

\_words[i].Visible = true;

\_words[i].LocationChanged += (sender, args) => this.Update();

}

}

private void MainForm\_Shown(object sender, EventArgs e)

{

foreach (var word in \_words)

{

\_wordsControllers.Add(new Word(word, Width, Height));

}

}

private void MainForm\_FormClosing(object sender, FormClosingEventArgs e)

{

\_wordsControllers.ForEach(elem => elem.Resume = false);

}

private void RestartButton\_Click(object sender, EventArgs e)

{

\_wordsControllers.RemoveAll(elem => true);

\_tryCounter = 1;

TryCounterValue.Text = \_tryCounter.ToString();

\_words.ForEach(elem => \_wordsControllers.Add(new Word(elem, Width, Height)));

RestartButton.Visible = false;

}

private void MainForm\_ResizeEnd(object sender, EventArgs e)

{

if (RestartButton.Visible) return;

\_wordsControllers.ForEach(elem =>

{

elem.Width = Width;

elem.Height = Height;

});

}

private void MainForm\_Click(object sender, EventArgs e)

{

if (RestartButton.Visible) return;

\_tryCounter++;

TryCounterValue.Text = \_tryCounter.ToString();

}

}